

PROGRAMACIÓN FUNCIONAL

Junio de 2010

Nombre y Apellidos:

1/1.5	2/1.5	3/2.5	4/1.5	5/3	Nota

1.- **(1.5 puntos)** Definir una función `eliminaAlgunos: [num] -> [num]` que elimine todos los números `n` de una secuencia tales que `n` sea impar y el último número par que aparece antes que `n` es mayor que `n`. Por ejemplo:

? `eliminaAlgunos [1,2,3,4,5]`
`[1,2,3,4,5]`

? `eliminaAlgunos [4,3,15,8,3,7,1]`
`[4,15,8]`

? `eliminaAlgunos [5,4,3,2,1]`
`[5,4,2]`

? `eliminaAlgunos [1,4,7,2,6,3,7,7,5]`
`[1,4,7,2,6,7,7]`

? `eliminaAlgunos [8,2,3]`
`[8,2,3]`

? `eliminaAlgunos [42,1,3,5,7,9,11]`
`[42]`

? `eliminaAlgunos [2,8,3]`
`[2,8]`

? `eliminaAlgunos [5,5,42,5,5,5,2,5,5]`
`[5,5,42,2,5,5]`

2.- **(1.5 puntos)** Dar el tipo más general para cada una de las siguientes funciones

a) `f1 a xs = takeWhile (/= a) xs ++ [a]`

b) `f2 g h (x,y) = (g x, h y)`

c) `f3 g h (x:xs) = [h y | y<-xs, x<=y]`

3.- **(2.5 puntos)** Definir una función `r` (incluido su tipo más general) tal que:

`r b = [x0, x1, x2, x3, x4, ...]`

donde `x0=b`

`xn = 1/2 * (xn-1 + b/xn-1)` para `n > 0`

(a) sin utilizar estructuras cíclicas, y

(b) utilizando estructuras cíclicas. En este caso, describir cómo se evalúa la expresión

`r 5`

hasta obtener los primeros 4 valores.

4.- **(1.5 puntos)** Describe brevemente los siguientes conceptos ilustrándolos con un ejemplo.

(a) Polimorfismo

(b) Currificación

(c) Orden-superior

5.- (3 puntos) Supongamos definido el siguiente tipo algebraico

```
data ConCad a = Nil
              | Unit a
              | Cad (ConCad a) (ConCad a)
```

con el objetivo de que `Nil`, `Unit x` y `Cad xs ys` representen respectivamente las expresiones del tipo `List a` siguientes: `[]`, `[x]` y `xs++ys`. En particular, ciertas parejas expresiones, como por ejemplo:

```
Cad xs (Cad ys zs)    y    Cad (Cad xs ys) zs
Cad zs Nil            y    zs
```

debieran ser iguales. Se pide:

(a) Definir la instancia de la clase **Eq** concreta para el tipo de datos `ConadenaCad a` de manera que dichas expresiones sean iguales., es decir que funcione del siguiente modo:

```
> Cad Nil (Unit 'p') == Unit 'p'
True

> Cad (Unit 'p') (Cad (Unit 'r') (Unit 's')) ==
Cad (Cad (Unit 'p') (Unit 'r')) (Unit 's')
True

> Cad (Unit 'p') (Unit 'r') == Cad (Unit 'r') (Unit 'p')
False
```

(b) Definir una función **primero** (y dar su tipo) que obtenga el primer elemento de una cadena de tipo `ConCad a`.

(c) Definir una función **numElementos** (y dar su tipo) que calcule el número de elementos de una cadena de tipo `ConCad a`.